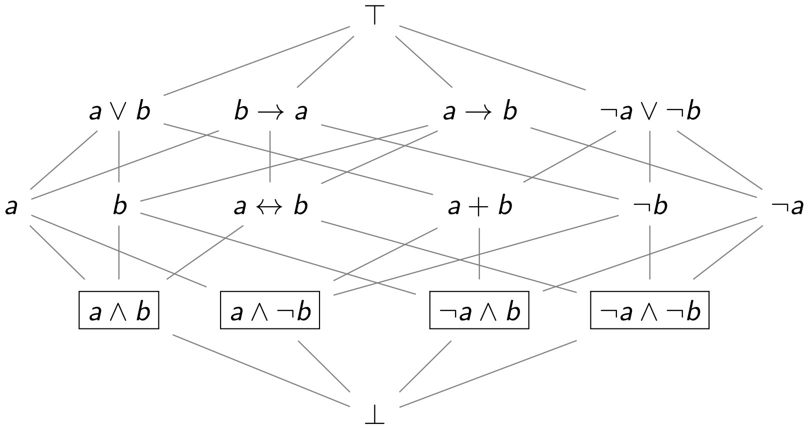
Weighted Model Counting with Conditional Weights for Bayesian Networks

Paulius Dilkas

27th November 2020

Boolean Algebras and Propositional Logic

Let $U = \{a, b\}$. Then 2^{2^U} is a Boolean algebra with the following Hasse diagram ($x \leq y$ if $x \subseteq y$ or, equivalently, $x = x \wedge y$).



Some Definitions

- A measure is a function $\mu\colon 2^{2^U} \to \mathbb{R}_{\geq 0}$ such that:
  - $\mu(\perp) = 0$;
  - $\mu(x \vee y) = \mu(x) + \mu(y)$ whenever $x \wedge y = \perp$.

Some Definitions

- A measure is a function $\mu \colon 2^{2^U} \to \mathbb{R}_{\geq 0}$ such that:
  - $\mu(\perp) = 0$;
  - $\mu(x \vee y) = \mu(x) + \mu(y)$ whenever $x \wedge y = \perp$.
- A weight function is any function $\nu \colon 2^U \to \mathbb{R}_{\geq 0}$.

Some Definitions

- A measure is a function $\mu\colon 2^{2^U} \to \mathbb{R}_{\geq 0}$ such that:
  - $\mu(\perp) = 0$;
  - $\mu(x \vee y) = \mu(x) + \mu(y)$ whenever $x \wedge y = \perp$.
- A weight function is any function $\nu\colon 2^U \to \mathbb{R}_{\geq 0}$.
  - It is factored if
    $$\nu = \prod_{x \in U} \nu_x$$
    for some functions $\nu_x\colon 2^{\{x\}} \to \mathbb{R}_{\geq 0}$, $x \in U$.

Some Definitions

- A measure is a function $\mu\colon 2^{2^U} \to \mathbb{R}_{\geq 0}$ such that:
  - $\mu(\bot) = 0$;
  - $\mu(x \vee y) = \mu(x) + \mu(y)$ whenever $x \wedge y = \bot$.
- A weight function is any function $\nu\colon 2^U \to \mathbb{R}_{\geq 0}$.
  - It is factored if
    $$\nu = \prod_{x \in U} \nu_x$$
    for some functions $\nu_x\colon 2^{\{x\}} \to \mathbb{R}_{\geq 0}$, $x \in U$.
- We say that ν induces μ if
  $$\mu(x) = \sum_{\{u\} \leq x} \nu(u)$$
  for all $x \in 2^{2^U}$.

Some Definitions

- A measure is a function $\mu\colon 2^{2^U} \to \mathbb{R}_{\geq 0}$ such that:
  - $\mu(\perp) = 0$;
  - $\mu(x \vee y) = \mu(x) + \mu(y)$ whenever $x \wedge y = \perp$.
- A weight function is any function $\nu\colon 2^U \to \mathbb{R}_{\geq 0}$.
  - It is factored if
  $$\nu = \prod_{x \in U} \nu_x$$
  for some functions $\nu_x\colon 2^{\{x\}} \to \mathbb{R}_{\geq 0}$, $x \in U$.
- We say that ν induces μ if
  $$\mu(x) = \sum_{\{u\} \leq x} \nu(u)$$
  for all $x \in 2^{2^U}$.
- A measure μ is factorable if there exists a factored weight function ν that induces μ.

WMC as a Measure on a Boolean Algebra

- Weighted model count (WMC) of a theory Δ, i.e.,

$$\mathrm{WMC}(\Delta) = \sum_{\omega \models \Delta} \prod_{\omega \models l} w(l)$$

  computes $\mu(x)$ for some $x \in 2^{2^U}$.

WMC as a Measure on a Boolean Algebra

- Weighted model count (WMC) of a theory Δ, i.e.,

$$\mathrm{WMC}(\Delta) = \sum_{\omega \models \Delta} \prod_{\omega \models l} w(l)$$

  computes $\mu(x)$ for some $x \in 2^{2^U}$.
- WMC with weights on literals can only compute factorable measures (c.f. independent probability distributions).
- Traditional workaround: expanding the Boolean algebra.

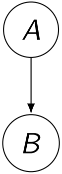
WMC as a Measure on a Boolean Algebra

- Weighted model count (WMC) of a theory Δ, i.e.,

$$\mathrm{WMC}(\Delta) = \sum_{\omega \models \Delta} \prod_{\omega \models l} w(l)$$

  computes $\mu(x)$ for some $x \in 2^{2^U}$.
- WMC with weights on literals can only compute factorable measures (c.f. independent probability distributions).
- Traditional workaround: expanding the Boolean algebra.
  - But we don't need to do that!
  - Instead, we can use conditional weight functions in the spirit of conditional probabilities.

Example: Encoding Bayesian Networks

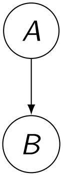


a	$\Pr(A = a)$
1	0.5
0	0.5

a	b	$\Pr(B = b \mid A = a)$
1	1	0.6
1	0	0.4
0	1	0.1
0	0	0.9

Figure: A Bayesian network with its conditional probability tables

Example: Encoding Bayesian Networks



a	$\Pr(A = a)$
1	0.5
0	0.5

a	b	$\Pr(B = b \mid A = a)$
1	1	0.6
1	0	0.4
0	1	0.1
0	0	0.9

Figure: A Bayesian network with its conditional probability tables

Let $U = \{\lambda_{A=1}, \lambda_{B=1}\}$. The weight function $\nu \colon 2^U \to \mathbb{R}_{\geq 0}$ for this network can be defined as $\nu := \nu_A \cdot \nu_B$, where $\nu_A = 0.5$, and

$$\begin{aligned}\nu_B = {} & 0.6[\lambda_{B=1}] \cdot [\lambda_{A=1}] + 0.4\overline{[\lambda_{B=1}]} \cdot [\lambda_{A=1}] \\ & + 0.1[\lambda_{B=1}] \cdot \overline{[\lambda_{A=1}]} + 0.9\overline{[\lambda_{B=1}]} \cdot \overline{[\lambda_{A=1}]}.\end{aligned}$$

Experimental Results